

Re: A chip too far? Where is your solution Mr Larkin?

```
;----- Variables -----  
;  
  
runSlow word -1 ; boolean, /F turns off  
hasFN word 0 ; boolean set with filename token  
fileHndl word ? ; MS-DOS File handle  
bitHdr1 bitFileHeader {  
  
;  
;----- Strings -----  
;  
  
msgFailFl byte "Invalid file name: $"  
msgFailFF byte 13,10  
byte "Invalid file format.",13,10,"$"  
msgFailBF byte "Bad file name or cannot open file.",13,10  
  
DataSeg ENDS  
  
;  
;----- Code begins here -----  
;  
  
CodeSeg SEGMENT  
  
ASSUME CS:CodeSeg,DS:DataSeg,ES:DataSeg  
cld  
push ds  
pop es ; copy DS to ES manually...  
  
;  
; Get own Program Segment Prefix and retrieve the first command line  
; parameter from it.  
;  
  
mov ah,62h  
int 21h ; get PSP in BX  
mov ch,0  
push ds  
  
ASSUME CS:CodeSeg,DS:nothing,ES:DataSeg  
mov ds,bx  
mov cl,ds:[80h] ; get and save length  
inc cx ; scan to one past the end  
; so the last token is valid  
mov si,81h ; DS:SI = command line  
  
...  
  
;  
;----- Procedures -----
```

Re: A chip too far? Where is your solution Mr Larkin?

```
;
;
; copyToken: finds the next token in the input string and copies it
; to an the output string.
; Input:
; DS:SI = input string location,
; ES:DI = output string location,
; CX = (unsigned) maximum number of bytes to scan.
; Make sure source has as much as CX bytes to scan and destination
; has as much as CX+1 free capacity allocated.
; Returns:
; DS:SI = delimiter which terminated this token,
; ES:DI = end of token copied,
; CX = number of bytes remaining. If CX = -1, copy ran out of
; bytes and the return string is probably incomplete.
;
copyToken PROC near USES ax
ASSUME CS:CodeSeg, DS:nothing, ES:nothing
jcxz loopOut ; trivial case
nextChar: lodsb ; get a character
call isDelimiter
jnz outChar
loop nextChar ; keep going until a token is found
jmp loopOut ; didn't find one, leave
outChar: ; found a token, copy it
stosb ; copy the first char
dec cx
jz loopOut
nextCharT: lodsb
call isDelimiter ; keep going until a delimiter
jz outCopyT ; is found
stosb
loop nextCharT
jmp loopOut
outCopyT: dec si
outCopy: ret

loopOut: mov cx,-1 ; loop fell through, set CX = -1
jmp outCopy

copyToken ENDP

....

CodeSeg ENDS

END

--==--==--
```

Re: A chip too far? Where is your solution Mr Larkin?

Re: A chip too far? Where is your solution Mr Larkin?

Much easier to read, if I do say so myself.

Tim

--

Deep Friar: a very philosophical monk.

Website: <http://webpages.charter.net/dawill/tmoranwms>

.